Tle

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/006,200	12/04/2001	William Charles Aegerter	33314/1:3	8843

7590 10/11/2006

Micah D. Stolowitz
Stoel Rives LLP
Suite 2600
900 SW Fifth
Portland, OR 97204

EXAMINER	
PATEL, MANGLESH M	
ART UNIT	PAPER NUMBER
2178	

DATE MAILED: 10/11/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	**Application No.** 10/006,200	**Applicant(s)** AEGERTER, WILLIAM CHARLES	
	Examiner Manglesh M. Patel	**Art Unit** 2178	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *Amendment (July 11, 2006)*.

2a)☒ This action is **FINAL**. 2b)□ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-11 and 13-19* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)□ Claim(s) *1-11& 13-19* is/are rejected.

7)□ Claim(s) _____ is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)□ The drawing(s) filed on _____ is/are: a)□ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) □ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) □ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application

6) □ Other: _____.

DETAILED ACTION

1. This **FINAL** action is responsive to the amendment filed on 07/11/06.

2. In the Amendment Claims 1-11 & 13-19 are pending. Claim 12 has been canceled. Claims 1 and 10 are independent claims.

Withdrawn Rejections

3. The 35 U.S.C. 101 rejections of claims 10-19 have been withdrawn in light of the amendment.

Claim Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5. Claims 1-11 & 13-19 remain rejected under 35 U.S.C. 103(a) as being unpatentable over Gupta (U.S. 6,513,059, filed August 24, 2000) in view of Sundaresan (U.S. 6,487,566, filed Oct 5, 1998).

Regarding Independent claim 1, Gupta teaches a computer-implemented, incremental process for executing an application servo in a client device based on a specified set of matching criteria, the process comprising the steps of: Selecting a servo to provide services (column 2, lines 31-52, wherein the servo is described by the agent since it describes a software component, application/domain components and system service); Identifying a data source associated with the selected servo (column 2, lines 31-52, wherein a data source is associated with the agent, the data source includes application/domain components); Initializing an execution context tree structure by creating a root node of the context tree associated with an initial instruction of the servo (column 2, lines 60-67 & column 3, lines 1-35, see fig 2 and column 6, lines 4-45, wherein the Awit space which is a collaborative environment for the agents is described by a context tree structure. Awit Space provides for the replication of context node(s) to support redundancy and scalability needs, resulting in a distributed system that supports a fluid configuration of intelligent agents or servos and context nodes where various operations are performed. Therefore the servos are described by the context tree structure inherently including an initialization of a root node);

Executing an instruction of the servo associated with the chosen context (column 4, lines 25-35, wherein agents communicate via context nodes, therefore instructions of the agent are executed based on the chosen context node); Responsive to said executing step, creating zero or more new child contexts in the context tree, each new child context including content defining a current internal evaluation state of the process (column 11, lines 39-54, wherein zero or more child contexts are created and they including a node controller thereby including content for defining an internal evaluation state of the process); Although Gupta teaches the use of context nodes for representing the agents that are selected based on rules he fails to teach the selection of a node based on a matching criteria. Sundaresan teaches choosing a context of the context tree that satisfies the matching criteria (See summary & column 5, lines 40-50, wherein rules described in XML use pattern matching and upon matching specified functions are performed. The entire node structure is represented by xml and the rules include a matching criteria used to perform different functions including transformations); And repeating said choosing, executing and creating steps over subsequent instructions of the servo until no context satisfies the matching criteria (See figure 3, wherein the matching step is repeated until no matches occur for the nodes resulting in the writing of the xml output). Gupta discloses the following: Responsive to changes to the data source, marking dependent contexts as unverified ; Choosing a marked content of the context tree ; Performing an instruction of the servo associated with the chosen context ; Responsive to said executing step, creating zero or more new child contexts in the context tree and removing or modifying zero or more existing child contexts ; Unmarking the chosen context and marking zero or more dependent contexts as unverified ;Repeating said choosing, performing, creating, removing , modifying, unmarking and marking steps over subsequent instructions of the servo until no contexts are left marked (abstract & column 6, lines 10-60, wherein Gupta discloses exchanging of information done by the intelligent agents to the nodes include information describing the state of a node. For example changes in a data source that are updated by the intelligent agents to the nodes include adding and deleting of the context tree. The Awit identifies the differences between the nodes). Although Gupta doesn't explicitly describe a unverified and marking state of the nodes, it would have been obvious to one of ordinary skill in the art at the time of the invention to have processed each node according to the rules till all nodes were updated. The motivation for doing so would have been to allow the Awit to handle processing of the nodes according to the specified rules within a changing environment thereby maintaining a consistent knowledge representation. Furthermore it would have been obvious to combine the teachings of Sundaresan with Gupta for the benefits of allowing the selection of nodes associated with a matching criteria thereby allowing the retrieval of content from the agent based on user interest.

Regarding Dependent claim 2, with dependency of.claim 1, Gupta teaches *wherein the content of the child context includes: a pointer to an element within the selected servo* (column 4, lines 25-35, wherein agents communicate via context nodes, therefore instructions of the agent are executed based on the chosen context node inherently including a pointer to the servo or agent); And a pointer that identifies a current data context by pointing into a source tree (column 12, lines 38-55, wherein a current node is identified upon request thereby inherently including a pointer to identify the currently selected node).

Regarding Dependent claim 3, with dependency of claim 1, Gupta teaches wherein the content of the child context includes: a reference to a parent context (column 6, lines 10-20, wherein the node framework maintains the relationship between the nodes, if the child context node did not keep track of the parent node then their would be no tree structure, therefore the relationship between the nodes are maintained by the definition of a tree structure); An ordered, potentially sparse, list of pointers to zero or more child contexts (column 11, lines 39-54, A tree must have pointers to a child node to be able to access the node context and that depends on the number of child nodes either zero or more. In addition to be able to call the node a child node it would inherently include a pointer to define the tree); and definitions for any symbols introduced by the context (column 5, lines 6-30, wherein the node context are described by the definitions).

Regarding Dependent claim 4, with dependency of claim 1, Gupta discloses responsive to said executing and performing steps, creating zero or more child spacers in the context tree representing unmaterialized child contexts (column 7 lines 55-67 & column 8, lines 1-11, wherein the policies are associated with a node context, thereby allowing the implementation of spacers to avoid certain node contexts by following the defined node policies); and wherein said choosing a context includes choosing either a context or a spacer (column 7 lines 55-67 & column 8, lines 1-11, wherein the selection of a context node also includes the associated policy of that node, thereby selection to apply changes to a node includes the direct selection of a node or by changing a policy associated with a context node).

Regarding Dependent claim 5, with dependency of claim 4, Gupta discloses wherein the context tree is implemented using a relative b-tree structure, and each spacer is reflected in an interior node entry in the relative b-tree structure to facilitate searching unmaterialized contexts (column 7 lines 55-67 & column 8, lines 1-11, Although Gupta doesn't specifically mention the type of tree structure used, he does disclose the use of various tree structures that would inherently include a b-tree structure since it defines a hierarchy type structure to maintain the context trees. The spacer is defined by the policy associated to the node context thereby allowing search operations for manipulating the tree structure using the node run-time framework).

Regarding Dependent claim 6, with dependency of claim 1, Gupta discloses wherein the b-tree node entry includes a field to track a linear value associated with a graphical display output object (column 6, lines 10-60, wherein the node framework is used to maintain the tree structure thereby including a linear value associated with a graphical display output based on the type of tree structure used).

Regarding Dependent claim 7, with dependency of claim 1, Gupta discloses wherein the process creates and maintains both the context tree and a geometry tree, the geometry tree representing the spatial structure of a predetermined graphical user interface (column 3, lines 15-35, wherein the Awit implements various interface functions. The node framework maintains the relationship of the context tree including a geometry tree that defines a GUI).

Regarding Dependent claim 8, with dependency of claim 1, Gupta discloses wherein the servo is defined using a servo definition language that references XML schema definitions as its core vocabulary (column 5, lines 53-61 & column 10, lines 13-23, wherein the servo or agent is described through the Awit communication in multiple transports including XML, thereby including XML schema definitions for defining the rules associated with the context tree).

Regarding Dependent claim 9, with dependency of claim 8, Gupta teaches a system for exchanging information over a network and describing the process using a context tree structure (See Abstract). Although Gupta describes that the agent or servos applications include descriptions in XML he fails to teach the use of data schema and transformation rules with the language. Sundaresan teaches wherein the servo definition language comprises: application data schema (column 7, wherein a DTD is used to describe the schema); transformation rules (See Abstract); and opportunity rules (See summary). At the time of the invention it would have been obvious to a person of ordinary skill in the art to include a pattern matching criteria in the rule definitions for the context nodes. The motivation for doing so would have been to allow the matching of content from a service provider based on the users interest. Therefore it would have been obvious to combine the teachings of Sundaresan with Gupta for the benefits of allowing the selection of nodes associated with a matching criteria thereby allowing the retrieval of content from the agent based on user interest.

Regarding Independent claim 10, Gupta teaches an interpreter stored in a computer-readable medium, the interpreter for interpreting a servo definition language for defining a distributed application that supports disconnected operation, the language comprising the following types of rules: Gupta teaches a system for exchanging information over a network and describing the process using a context tree structure (See Abstract). And interface object specifications (column 3, lines 15-35, wherein the Awit implements various interface functions, thereby including interface object specifications). Although Gupta describes that the agent or servos applications include descriptions in XML including rules he fails to teach the use of data schema and transformation rules with the language. Sundaresan teaches application data schema (column 7, wherein a DTD is used to describe the schema); Transformation rules (See Abstract); Transaction handling rules (See Abstract); Gupta describes the following: Opportunity rules to realize automatic extension or integration of servos through opportunity-based linking of an interface component representing an instance of a schema fragment to a template (column 6, lines 35-67, Gupta discloses that by using ASpeak the agents are able to communicate with each other based on the defined rules of the agents). Gupta suggests rules while Sundaresan describes the rules defined in a schema and DTD for pattern matching. At the time of the invention it would have been obvious to a person of ordinary skill in the art to include a pattern matching criteria in the rule definitions for the context nodes. The motivation for doing so would have been to allow the matching of content from a service provider based on the users interest. Therefore it would have been obvious to combine the teachings of Sundaresan with Gupta for the benefits of allowing the selection of nodes associated with a matching criteria thereby allowing the retrieval of content from the agent based on user interest.

Regarding Dependent claim 11, with dependency of claim 10, Gupta discloses an interpreter comprising access rules (column 2, lines 30-60, wherein the Awit space that provides the collaboration environment for the agents or servos include the support for secure distributed application model that define the access rules).

Regarding Dependent claim 13, with dependency of claim 10, Gupta teaches a system for exchanging information over a network and describing the process using a context tree structure (See Abstract). And interface object specifications (column 3, lines 15-35, wherein the Awit implements various interface functions, thereby including interface object specifications). Although Gupta describes that the agent or servos applications include descriptions in XML including rules he fails to teach the use of data schema and transformation rules with the language. Sundaresan teaches wherein the template specifies at least one of a transformation rule (See Abstract), a transaction handling rule and an interface object specification. Therefore it would have been obvious to combine the teachings of Sundaresan with Gupta for the benefits of allowing the selection of nodes associated with a matching criteria thereby allowing the retrieval of content from the agent based on user interest.

Regarding Dependent claim 14, with dependency of claim 10, Gupta teaches an abstract interface object definition (column 3, lines 15-35, wherein the Awit implements various interface functions, thereby including interface object definitions).

Regarding Dependent claim 15, with dependency of claim 10, Gupta teaches a system for exchanging information over a network and describing the process using a context tree structure (See Abstract). Although Gupta describes that the agent or servos applications include descriptions in XML he fails to teach the use of data schema and transformation rules with the language. Sundaresan teaches a servo definition language wherein the application data schema comprises an XML-based schema (column 7, wherein a DTD is used to describe the schema in XML). At the time of the invention it would have been obvious to a person of ordinary skill in the art to include a pattern matching criteria in the rule definitions for the context nodes. The motivation for doing so would have been to allow the matching of content from a service provider based on the users interest. Therefore it would have been obvious to combine the teachings of Sundaresan with Gupta for the benefits of allowing the selection of nodes associated with a matching criteria thereby allowing the retrieval of content from the agent based on user interest.

Regarding Dependent claim 16, with dependency of claim 10, Gupta discloses an interpreter defined using XML schema definitions XSD as the core vocabulary. (column 5, lines 53-61 & column 10, lines 13-23, wherein the servo or agent is described through the Awit communication in multiple transports including XML, thereby including XML schema definitions for defining the rules associated with the context tree).

Regarding Dependent claim 17, with dependency of claim 10, Gupta teaches a system for exchanging information over a network and describing the process using a context tree structure (See Abstract). And interface object specifications (column 3, lines 15-35, wherein the Awit implements various interface functions, thereby including interface object specifications). Although Gupta describes that the agent or servos applications include descriptions in XML including rules he fails to teach the use of data schema and transformation rules with the language. Sundaresan teaches a servo definition language including a view element for selecting a group of the said transformation rules to define at least a part of an output interface (See Abstract). At the time of the invention it would have been obvious to a person of ordinary skill in the art to include a pattern matching criteria in the rule definitions for the context nodes. The motivation for doing so would have been to allow the matching of content from a service provider based on the users interest. Therefore it would have been obvious to combine the teachings of Sundaresan with Gupta for the benefits of allowing the selection of nodes associated with a matching criteria thereby allowing the retrieval of content from the agent based on user interest.

Regarding Dependent claim 18, with dependency of claim 10, Gupta teaches a servo definition language including a storage declaration element that enables an author to reserve and name persistent storage for use by the servo and any other servos authorized to access the corresponding data (column 2, lines 30-67, wherein the servo language is described using XML and the Awit space is used with the servo or agent to provide a collaborative environment that includes access by client authorization including storage declarations provided by defining the context nodes associated with the agent).

Regarding Dependent claim 19, with dependency of claim 18, Gupta teaches the a servo definition language wherein the storage declaration element includes a locally scoped name for a corresponding storage tree and identifies a schema to which the storage tree must conform (column 2, lines 30-67 & see Abstract, wherein the servo language is described using XML and the Awit space is used with the servo or agent to provide a collaborative environment that includes access by client authorization including storage declarations provided by defining the context nodes associated with the agent. The storage declarations include the identification of a schema or rule associated with the context node).

It is noted that any citation [[s]] to specific, pages, columns, lines, or figures in the prior art references and any interpretation of the references should not be considered to be limiting in any way. A reference is relevant for all it contains and may be relied upon for all that it would have reasonably suggested to one having ordinary skill in the art. [[See, MPEP 2123]]

Response to Arguments

6. Applicant's arguments filed July 11, 2006 have been fully considered but they are not persuasive.

The Applicant Argues: <u>Gupta does not disclose or suggest the process recited in claim 1 for executing an application servo" using a "context tree".</u> (Page 8, paragraph 1)

However the examiner respectfully disagrees. Gupta describes the use of "Awit" that are an autonomous, mobile, intelligent, collaborative and adaptive software program that acts as a digital proxy for the client or user. The Awit works in a live environment, a distributed, context-based knowledge space, where content is evolving constantly (column 6, lines 37-62). The applicant states that the word context means an "internal evaluation state of the process". The Awit works with the context trees for updating the nodes. These agents determine weather nodes should be added or deleted (column 6, lines 10-35). Further the agents communicate and understand the differences (internal evaluation state) between the context nodes such as modifications to the state of a node. Further Gupta discloses the amended portions as described above in the office action. The amended portion to claim 10 describes rules that allow the servos to communicate between other servos. Gupta teaches that the Awit communicates with other Awit's to extend the functionality of the applications and tasks (column 6, lines 45-67).

(Note: The claim fail to clearly identify the inventive feature pertaining to the disconnected operation of mobile devices. Most of the language in the claim describes processes for executing the servo but fail to include how the servo operation relates to disconnected mobile devices. The examiner is open to interviews if the applicant is willing to expedite prosecution of the application via amendment to the claims, otherwise only written responses will be considered in an after final amendment).

Conclusion

7. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Manglesh M. Patel whose telephone number is (571) 272-5937. The examiner can normally be reached on M,F 8:30-6:00 T,TH 8:30-3:00 Wed 8:30-7:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Stephen S. Hong can be reached on (571)272-4124. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Manglesh M. Patel
Patent Examiner
September 23, 2006

CESAR PAULA
PRIMARY EXAMINER